Network Reconstruction and Analysis: TP2

Mathieu Pont and Lucas Rodrigues Pereira

December 2019

Abstract

1 Introduction

In the context of the Master's Degree on Machine Learning for Data Science, at the University of Paris (Descartes), we have been given the task to discuss some network reconstruction methods, as part of the "Network Reconstruction and Analysis" course. The main idea of network reconstruction is to find causality between variables through a graph.

We used the *insurance* dataset proposed by [Binder, 1997]. It having the aim of evaluating car insurance risk according to 27 qualitative variables on 20,000 individuals.

2 Experiments and Results

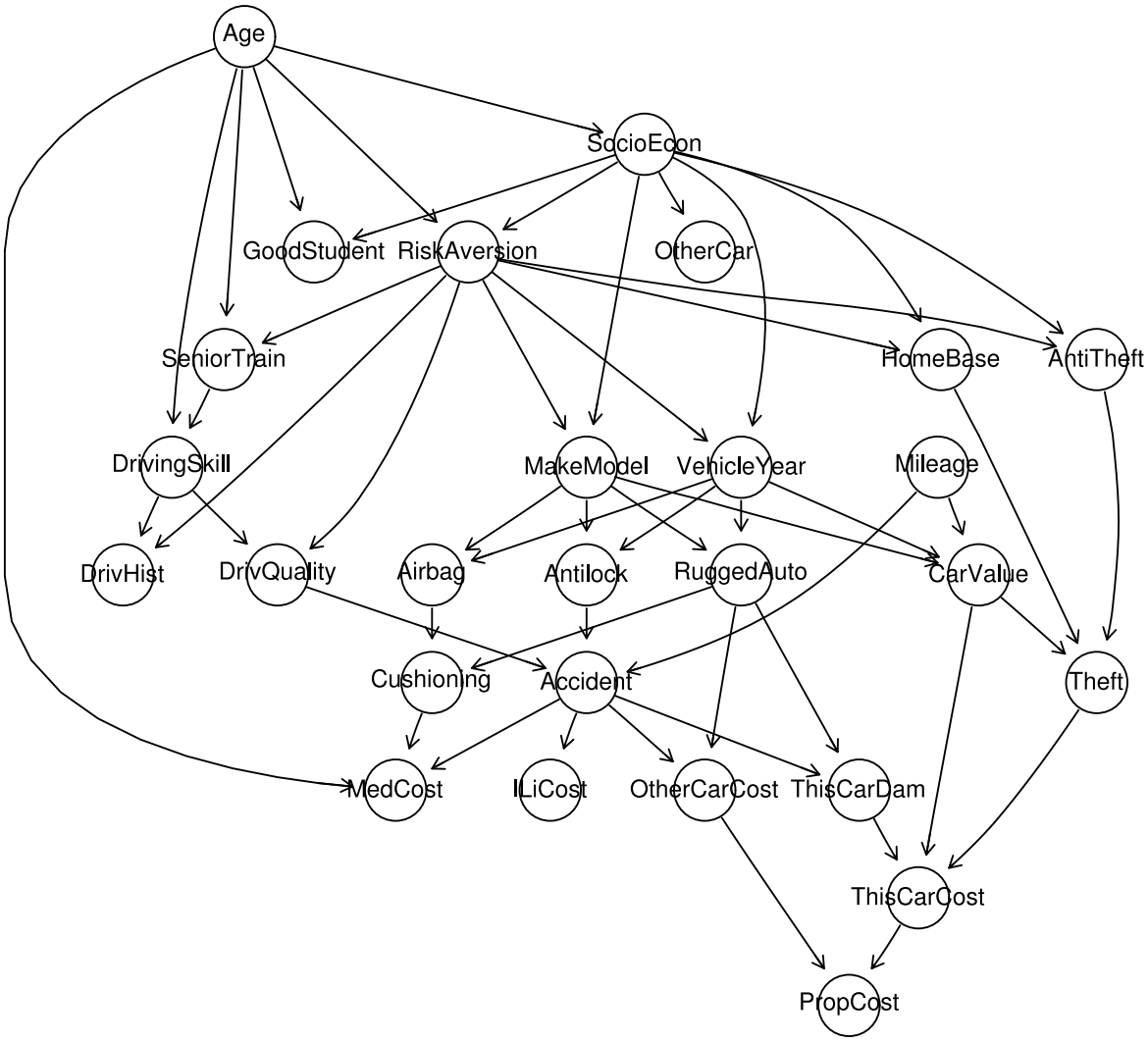

Figure 1: Ground truth model.

In Figure 1 we see the ground truth graph for the *insurance* dataset. We can see, for example, that the variable "Age" seems to influence the variables "SocioEcon" (socio-economic status), "DrivingSkill" and "MedCost" (cost of the medical treatment). We can easily see how these three variables can be influenced by the age of the person. "Accident" (severity of the accident) seems to influence "MedCost" too, which makes sense since the cost of medical treatment is closely related to severity of the accident.

We have used three methods to reconstruct the network based on the dataset: Hill-Climbing (search & score), PC (constraint based) and Aracne (information-theoretic). The following three

graphs have been reconstructed using each one of these method respectively.

False positive edges (appearing in the predicted graph but not in the real one) are highlighted in red but are taken into account only FP edges of skeleton graphs (representation of a graph where we do not keep the direction of the edges).

2.1 HC

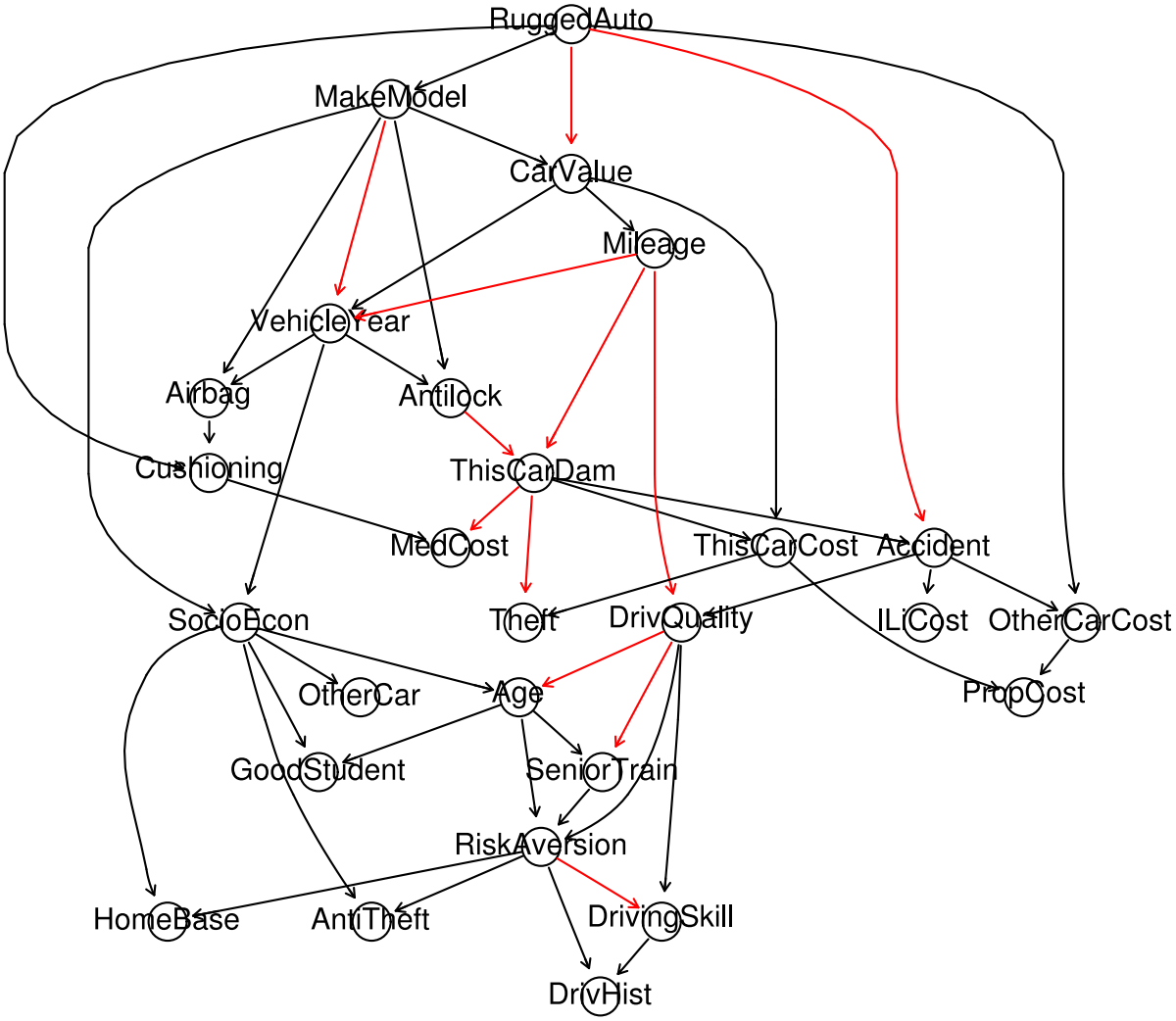

Figure 2: Reconstructed network using the HC algorithm.

True Positive	False Positive	False Negative	Precision	Recall	Fscore
76	24	28	0.76	0.73	0.75

Table 1: Metrics for the HC algorithm for the skeleton graph.

As seen in Table 1 the reconstructed network with the HC algorithm have pretty good performances based on skeleton graphs. However, many edges are found whereas they should not be (FP) and many are not found at all (FN) but the metrics are balanced.

If we had computed FP edges directly with the predicted graph, edges appearing in both graphs (predicted and ground truth) but in different direction would be considered as FP. That means that, when we used skeletons, FP edges appearing in the graph can not appear even in the other direction in the ground truth graph (because we remove the notion of direction with skeleton). The differences in metrics for both methods are discussed in 2.3.

Then we take the FP edges of the skeleton graph and highlight them in the predicted graph making their number to 12:

$\boldsymbol{ThisCarDam \rightarrow MedCost}$	$\boldsymbol{DrivQuality \rightarrow Age}$	$\boldsymbol{DrivQuality \rightarrow SeniorTrain}$
$\boldsymbol{RuggedAuto \rightarrow Accident}$	$\boldsymbol{Mileage \rightarrow DrivQuality}$	$\boldsymbol{Antilock \rightarrow ThisCarDam}$
$\boldsymbol{Mileage \rightarrow VehicleYear}$	$\boldsymbol{MakeModel \rightarrow VehicleYear}$	$\boldsymbol{ThisCarDam \rightarrow Theft}$
$\boldsymbol{RuggedAuto \rightarrow CarValue}$		$\boldsymbol{RiskAversion \rightarrow DrivingSkill}$
$\boldsymbol{Mileage \rightarrow ThisCarDam}$		
Can make sense	*Seems to not make sense*	*Don't know*

Table 2: FP edges for the HC algorithm.

Some FP edges seem incoherent according to common sense. For example the edge $\boldsymbol{DrivQuality \rightarrow Age}$ is supposed to tells us that the quality of driving should has an influence on the age of someone, but it seems to be the other way around.

However, other FP edges can make sense, for example, $\boldsymbol{ThisCarDam \rightarrow MedCost}$ tells us that the damage of the car can influences the cost of the medical treatment.

2.2 PC

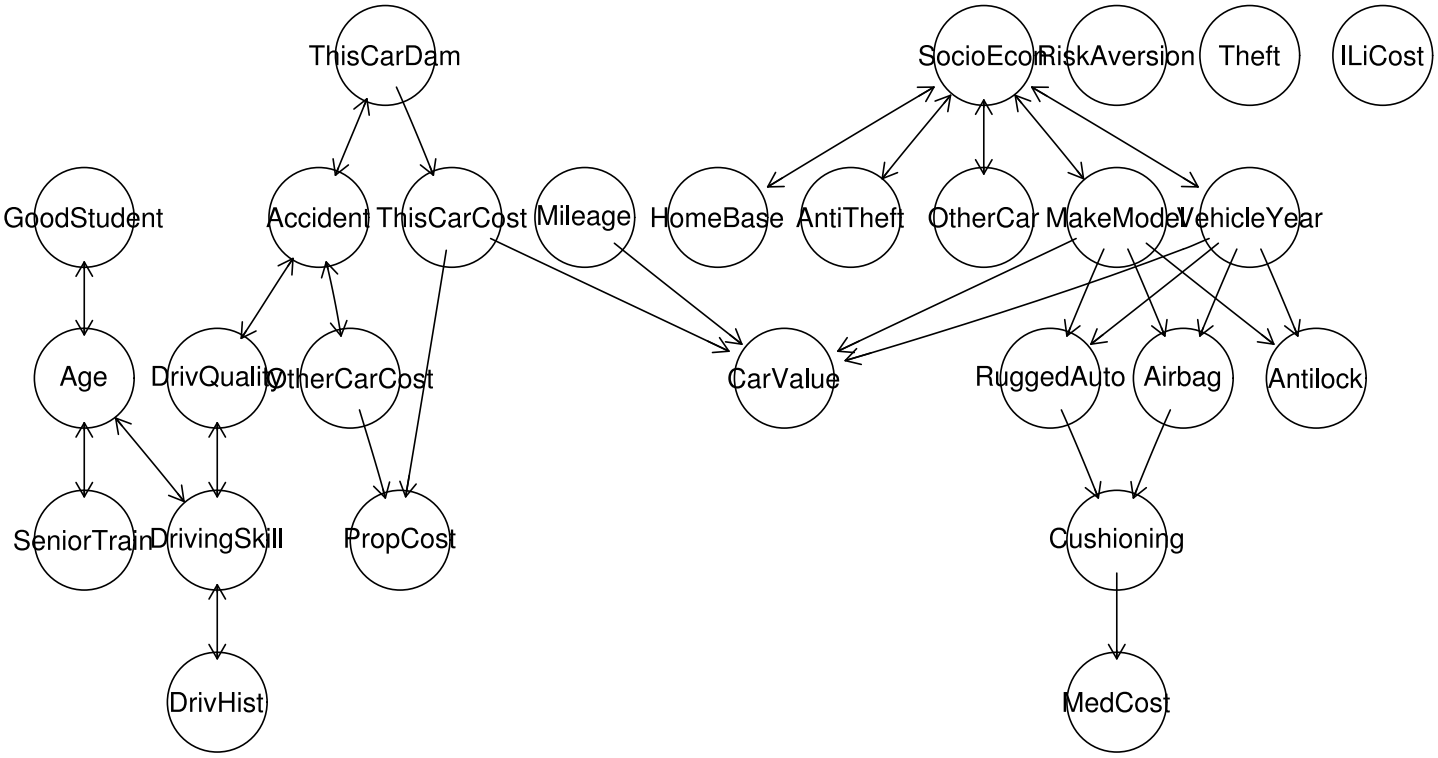

Figure 3: Network with PC algorithm.

True Positive	False Positive	False Negative	Precision	Recall	Fscore
58	0	46	1.00	0.56	0.72

Table 3: Metrics for the PC algorithm for the skeleton graph.

Table 3 shows us that the PC algorithm has a perfect precision but not a very good recall. Therefore, as we can see, there is no FP edges, meaning that PC does not find new edges compared to the ground truth graph. On one hand it is nice since we do not want the algorithm to be wrong but on the other hand it can't discover meaningfull new relations as HC did.

2.3 Aracne

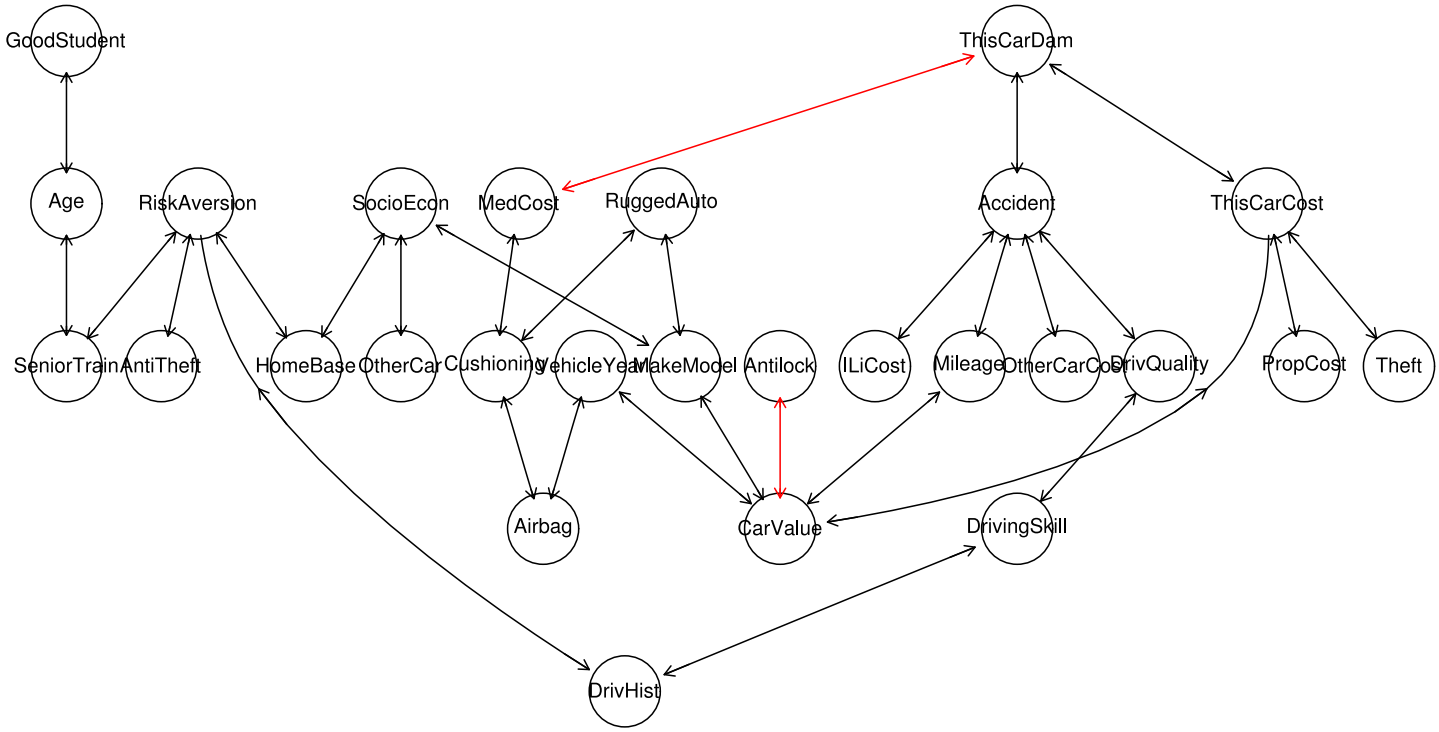

Figure 4: Network with Aracne algorithm for the skeleton graph.

We must notice that the graph is actually undirected since we use Aracne which does not provide direction of edges.

True Positive	False Positive	False Negative	Precision	Recall	Fscore
56	4	48	0.93	0.54	0.68

Table 4: Metrics for the Aracne algorithm.

Aracne's performance is similar to that of PC. It has a very high precision but not a good recall. Here we have only 2 FP edges:

$$\boldsymbol{ThisCarDam} \leftrightarrow \boldsymbol{MedCost} \;\Big|\; \boldsymbol{Antilock} \leftrightarrow \boldsymbol{CarValue}$$

The first one is also found by HC and as it was already discussed, the link between these variables make sense (damage of the car and medical treatment cost). The second one could make sense too, if a vehicle has antilock (ABS) system its value can be higher than a vehicle having not this option.

2.4 Comparison

	Predicted			Skeleton			Cpdag		
	HC	PC	Aracne	HC	PC	Aracne	HC	PC	Aracne
True Positive	26	14	**28**	**76**	58	56	33	**36**	**36**
False Positive	**24**	28	32	24	**0**	4	22	**7**	24
False Negative	26	38	**24**	**28**	46	48	37	**34**	**34**
Precision	**0.52**	0.33	0.47	0.76	**1.00**	0.93	0.60	**0.84**	0.6
Recall	0.50	0.27	**0.54**	**0.73**	0.56	0.54	0.47	**0.51**	**0.51**
Fscore	**0.51**	0.30	0.50	**0.75**	0.72	0.68	0.53	**0.64**	0.55

Table 5: Metrics for each method, comparison between predicted graphs (left), skeleton graphs (middle) and cpdag graphs (right).

In table 5 we see the value of each metric for each algorithm and representation. We also compared graphs as cpdag (Completed Partially Directed Acyclic Graph) which can be seen as a trade-off between "predicted" (we do not add edges) and "skeleton" (we add all edges in the other direction).

We see that HC is very well balanced for each representation. PC tends to have a good precision but suffers in term of recall.

It is interesting to see that, for the predicted graph, PC has a very low value in term of precision compared to the one for the skeleton graph. According to the FP value, that means that PC found a lot of edges of the original graph but the direction were not correct for many of them. These edges are therefore not FP with the skeleton graph since it is not taking into account the direction.

The same comment can be made for Aracne, moreover we see why it is important to use skeleton for this algorithm, the edges are not directed (treated as an edge in each direction) therefore even if the algorithm finds a correct edge, the number of FP will increase because the original graph has the edge in only one direction.

On the contrary, HC has the same amount of FP edges for the predicted and for the skeleton graphs meaning that this number is not due to the direction of the edges.

They all have their strength and weaknesses and it is complicated to select one method as the best. Nevertheless, PC seems to outperform the other algorithms with the cpdag representation.

3 Conclusion

The network reconstruction based on data is very relevant to understanding the causality relation among variables. However, it is not always possible and often inaccurate. Good results can be found, and there is plenty of research opportunity in the field.

In this work, we have compared the performance of three methods: Hill-Climbing (search & score), PC (constraint based) and Aracne (information-theoretic). All three have performed relatively well in finding relations among variables. They have, notwithstanding, struggled to find causality (in the graphs, represented by the direction of the edges).

References

[Binder, 1997] Binder, J. (1997). Adaptive probabilistic networks with hidden variables.